CHRP™ ISA Keyboard/Mouse Controller Device Binding

to:

IEEE 1275-1994

Standard for Boot

(Initialization, Configuration)

Firmware

Revision: 1.0 Unapproved DRAFT

Date: May 6, 1996

Table of Contents

1. Purpose of this Device Binding 3
2. Revision History 3
3. References 3
4. Definition of Terms 3
5. Device Characteristics (Informative) 3
5.1 Address Formats and Representations 3
5.1.1 Physical Address Format: Numerical Representation 3
5.1.2 Physical Address Format: Text Representation 4
6. Device-specific Configuration Variables 4
7. Device Nodes 4
7.1 Properties 4
7.1.1 Open Firmware-defined Properties for Device Nodes 4
7.1.2 Device-specific Properties for Device Nodes 5
7.2 Methods 5
7.2.1 Open Firmware-defined Methods for Device Nodes 5
7.2.1.1 Device Arguments for “Open” Method 5
7.2.2 Device-specific Methods for Device Nodes 5
8. Child Nodes 5
8.1 Properties 5
8.1.1 Open Firmware-defined Properties for Child Nodes 5
8.1.2 Device-specific Properties for Child Nodes 6
8.1.3 Device-specific Methods for Child Nodes 6
9. User Interface Commands 6
9.1 Open Firmware-defined User Interface Commands 6
9.2 Device-specific User Interface Commands 6
10. Device State 6
10.1 Device State When Client is Started 6
10.2 Device State Required When Client Calls Open Firmware 6
11. Other Commentary 7

1. Purpose of this Device Binding

This document specifies the application of Open Firmware to the PowerPC Common Hardware Reference Platform (CHRP™) ISA keyboard/mouse controller, including device-specific requirements and practices for initialization, properties, and methods. This device shall be a child of an ISA or EISA Bus Node.

2. Revision History

Revision 1.0 Unapproved DRAFTMay 6, 1996Initial revision. Jordan Brown, Sunsoft and John Kingman, IBM editors

3. References

This Open Firmware System binding standard *shall* be used in conjunction with the following publications. When the following standards are superseded by an approved revision, the revision *shall* apply.

[1] *IEEE Std 1275-1994 Standard for Boot (Initialization, Configuration) Firmware, Core Practices and Requirements*.

[2] *Core Errata, IEEE P1275.7/D4.*

[3] *ISA/EISA/ISA-PnP binding to: IEEE Std 1275-1994, Standard for Boot (Initialization, Configuration) Firmware*.

[4] *Device Support Extensions to: IEEE Std 1275-1994, Standard for Boot (Initialization, Configuration) Firmware*.

[5] *PowerPC Microprocessor Common Hardware Reference Platform: I/O Device Reference*. This document describes the PowerPC Common Hardware Reference Platform (CHRP™) System Standard I/O Devices; hardware registers, register locations, and hardware attributes.

[6] *Open Firmware Recommended Practice: Interrupt Mapping*.

[7] *Open Firmware Recommended Practice: Generic Names*.

[8] *PowerPC Microprocessor Common Hardware Reference Platform binding to: IEEE Std 1275-1994, Standard for Boot (Initialization, Configuration) Firmware*.

4. Definition of Terms

EISA: Extended Industry Standard Architecture

ISA: Industry Standard Architecture

5. Device Characteristics (Informative)

The keyboard/mouse controller is actually a microcontroller (commonly based on an 8042 microcontroller) which monitors the keyboard and mouse for input, provides data through a separate I/O port for the keyboard and mouse and signals that such data is available with an interrupt.

5.1. Address Formats and Representations

This section describes the physical address representation.

5.1.1. Physical Address Format: Numerical Representation

The numerical representation of a keyboard or mouse address is a binary integer in the range ’0’ to ’1’.

5.1.2. Physical Address Format: Text Representation

The textual representation of a keyboard or mouse address is an ASCII “0” or an ASCII “1.”

6. Device-specific Configuration Variables

None.

7. Device Nodes

7.1. Properties

As specified in [1], [3] and [8], with the following additions or modifications. This node shall not have a “`ranges`” property.

7.1.1. Open Firmware-defined Properties for Device Nodes

`"name"` **S**

Standard *property name*, specifies the generic name of the device.

The meaning of this property is as defined in Open Firmware core document [1], as modified by the *Generic Names Recommended Practice*[7]. The value for nodes described by this specification shall be **`"`**`8042`**`"`**.

`"device_type"` **S**

Standard *property name* to define the device’s implemented interface.

The meaning of this property is as defined in the Open Firmware core document [1]. The value for nodes described by this specification shall be “`8042`”.

`"compatible"` **S**

Standard *property name*, specifies device names with which this device is compatible.

The meaning of this property is as defined in Open Firmware, as modified by the *Generic Names Recommended Practice* [7]. As described in those documents, the entries are a list of device names with which this device is compatible, starting with the name of the device itself and progressing through successively less precise and possibly less functional compatible devices.

The value of this property shall include “`chrp,8042`”

Additional entries may be supplied, at their appropriate position in the list, to describe devices with which this device is compatible.

`"reg"` **S**

Standard *property name* to define the package’s registers.

The meaning of this property is as defined in the Open Firmware core document [1]. It describes the device's register set. The values which shall be assigned to this property are explained in the *ISA/EISA/ISA-PnP binding*[3] and the *I/O Device Reference*[5].

`"interrupts"` **S**

Standard *property name* to define a package’s interrupts.

The meaning of this property is as defined in the *Interrupt Mapping Recommended Practice* [6]. The values assigned to this property are explained in the *ISA/EISA/ISA-PnP binding*[3] and the *I/O Device Reference*[5]. This device may have up to two interrupts assigned to it. The components of the array shall correspond with the interrupts assigned to the children of this node such that a child’s reg property when used as an index into the array shall yield the appropriate interrupt value for that child.

`"#address-cells"` **S**

Standard *prop-name* to define the number of cells required to represent the physical addresses for the children of this node.

prop-encoded-array: Integer constant 1, encoded as with **encode-int**.

The value of "`#address-cells`" for this node *shall* be `1`.

"#size-cells" **S**

Standard *prop-name* to define the number of cells necessary to represent the length of a physical address range.

prop-encoded-array: Integer constant 0, encoded as with **encode-int.**

The value of "`#size-cells`" for this node shall be 0 because the children of this node do not consume any physical address space.

7.1.2. Device-specific Properties for Device Nodes

None.

7.2. Methods

7.2.1. Open Firmware-defined Methods for Device Nodes

As specified in [1] and [3], without addition or modification.

7.2.1.1. Device Arguments for "Open" Method

As specified in [1] and [3], without addition or modification.

7.2.2. Device-specific Methods for Device Nodes

None.

8. Child Nodes

8.1. Properties

As specified in [1] and [3], with the following additions or modifications.

8.1.1. Open Firmware-defined Properties for Child Nodes

As specified in [1] and [3], with the following additions or modifications.

"name" **S**

Standard *property name*, specifies the generic name of the device.

The meaning of this property is as defined in Open Firmware core document [1], as modified by the Generic Names Recommended Practice. The value for nodes described by this specification shall be "`keyboard`" for an ISA keyboard device node, and "`mouse`" for an ISA mouse device node.

"device_type" **S**

Standard *property name* to define the device's implemented interface.

The meaning of this property is as defined in the Open Firmware core document [1]. The value for nodes described by this specification shall be "`keyboard`" for an ISA keyboard device node, or "`mouse`" for an ISA mouse device node.

"compatible" **S**

Standard *property name*, specifies device names with which this device is compatible.

The meaning of this property is as defined in Open Firmware, as modified by the Generic Names Recommended Practice. As described in those documents, the entries are a list of device names with which this device is compatible, starting with the name of the device itself and progressing through successively less precise and possibly less functional compatible devices.

The value of this property shall be "`pnpPNP,303`" for an ISA keyboard device node.

The value of this property shall be "`pnpPNP,f03`" for an ISA mouse device node.

Additional entries may be supplied, at their appropriate position in the list, to describe devices with which this device is compatible.

"reg" **s**

prop-name, specifies the number of this device.

prop-encoded-array: An integer, encoded with **encode-int**.

The value of this property shall be 0 for a keyboard node and 1 for a mouse node.

8.1.2. Device-specific Properties for Child Nodes

As specified in [4] without addition or modification.

8.1.3. Device-specific Methods for Child Nodes

As specified in [4] without addition or modification.

9. User Interface Commands

9.1. Open Firmware-defined User Interface Commands

None.

9.2. Device-specific User Interface Commands

None.

10. Device State

10.1. Device State When Client is Started

For devices not selected as Open Firmware's "console input device" or "console output device" device, the initial state is undefined. To comply with this binding, the device's outputs and interrupts shall be "tri-stated" when the client is started.

For devices which are selected as Open Firmware's "console input device" or "console output device" device, the device shall be initialized appropriately for the device to which it is attached.

Refer to [5] for more information on the state of this device when the client is started.

10.2. Device State Required When Client Calls Open Firmware

For devices not selected as Open Firmware's "console input device" or "console output device" device, there is no requirement.

For devices selected as Open Firmware's "console input device" or "console output device" device, the state should be unmodified from the initial state on client start-up.

Note: If the device is in a different state when the client calls Open Firmware, unpredictable behavior may result if Open Firmware accepts input or generates output. Clients changing the device state should either restore the original state before calling Open Firmware or should avoid using Open Firmware features requiring user interaction. Changing the device state is likely to render Open Firmware unusable for debugging purposes.

11. Other Commentary